WE CLAIM

1. Apparatus for processing data, said apparatus comprising:

(i) a processor core operable to execute operations as specified by instructions of a first instruction set, said processor core having an instruction pipeline into which instructions to be executed are fetched from a memory and along which instructions progress; and

(ii) an instruction translator operable to translate instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set; wherein

(iii) said instruction translator is within said instruction pipeline and translates instructions of said second instruction set that have been fetched into said instruction pipeline from said memory;

(iv) at least one instruction of said second instruction set specifies a multi-step operation that requires a plurality of operations that may be specified by instructions of said first instruction set in order to be performed by said processor core; and

(v) said instruction translator is operable to generate a sequence of translator output signals to control said processor core to perform said multi-step operation.

2. Apparatus as claimed in claim 1, wherein said translator output signals include signals forming an instruction of said first instruction set.

3. Apparatus as claimed in any one of claims 1 and 2, wherein said translator output signals include control signals that control operation of said processor core and match control signals produced on decoding instructions of said first instruction set.

4. Apparatus as claimed in any one of claims 1, 2 and 3, wherein said translator output signals include control signals that control operation of said processor core and specify parameters not specified by control signals produced on decoding instructions of said first instruction set.

5. Apparatus as claimed in any one of the preceding claims, wherein said processor core fetches instructions from an instruction address within said memory specified by a program counter value held by said processor core.

6. Apparatus as claimed in claim 5, wherein, when an instruction of said second instruction set is executed, said program counter value is advanced by an amount that is independent of whether or not said instruction of said second instruction set specifies a multi-step operation.

7. Apparatus as claimed in any one of claims 5 and 6, wherein, when an instruction of said second instruction set is executed, said program counter value is advanced to specify a next instruction of said second instruction set to be executed.

8. Apparatus as claimed in any one of claims 5, 6 and 7, wherein said program counter value is saved if an interrupt occurs when executing instructions of said second instruction set so and is used to restart execution of said instructions of said second instruction set after said interrupt.

9. Apparatus as claimed in any one of the preceding claims, wherein instructions of said second instruction set specify operations to be executed upon stack operands held in a stack.

10. Apparatus as claimed in any one of the preceding claims, wherein said processor has a register bank containing a plurality of registers and instructions of said first instruction set execute operations upon register operands held in said registers.

11. Apparatus as claimed in claim 10, wherein a set of registers within said register bank hold stack operands from a top potion of said stack.

12. Apparatus as claimed in claims 9 and 11, wherein said instruction translator has a plurality of mapping states in which different registers within said set of registers hold respective stack operands from different positions within said stack, said instruction translator being operable to move between mapping states in dependence upon operations that add or remove stack operands held within said stack.

13. Apparatus as claimed in any one of the preceding claims, further comprising a bypass path within said instruction pipeline such that said instruction translator may be bypassed when instructions of said second instruction set are not being processed.

14. Apparatus as claimed in any one of the preceding claims, wherein said instructions of said second instruction set are Java Virtual Machine bytecodes.

15. A method of processing data using a processor core having an instruction pipeline into which instructions to be executed are fetched from a memory and along which instructions progress, said processor core being operable to execute operations specified by instructions of a first instruction set, said method comprising the steps of:

(i) fetching instructions into said instruction pipeline; and

(ii) translating fetched instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set using an instruction translator within said instruction pipeline; wherein

(iii) at least one instruction of said second instruction set specifies a multi-step operation that requires a plurality of operations that may be specified by instructions of said first instruction set in order to be performed by said processor core; and

(iv) said instruction translator is operable to generate a sequence of translator output signals to control said processor core to perform said multi-step operation.

16. A computer program product holding a computer program for controlling a computer to perform the method of claim 13.

17. Apparatus for processing data, said apparatus comprising:

(i) a processor core operable to execute operations as specified by instructions of a first instruction set, said processor core having an instruction pipeline into which instructions to be executed are fetched from a memory and along which instructions progress; and

(ii) an instruction translator operable to translate instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set; wherein

(iii) said instructions of said second instruction set are variable length instructions;

said instruction translator is within said instruction pipeline and translates instructions of said second instruction set that have been fetched into a fetch stage of said instruction pipeline from said memory; and

(iv) said fetch stage of said instruction pipeline includes an instruction buffer holding at least a current instruction word and a next instruction word fetched from said memory such that if a variable length instruction of said second instruction set starts within said current instruction word and extends into said next instruction word, then said next instruction word is available within said pipeline for translation by said instruction translator without requiring a further fetch operation.

18. Apparatus as claimed in claim 17, wherein said instruction buffer is a swing buffer.

19. Apparatus as claimed in any one of claims 17 and 18, wherein said fetch stage includes a plurality of multiplexers for selecting a variable length instruction from one or more of said current instruction word and said next instruction word.

20. Apparatus as claimed in any one of claims 17, 18 and 19, wherein said instructions of said second instruction set are Java Virtual Machine bytecodes.

21. Apparatus as claimed in any one of claims 17 to 20, further comprising a bypass path within said instruction pipeline such that said instruction translator may be bypassed when instructions of said second instruction set are not being processed.

22. Apparatus as claimed in any one of claims 17 to 21, wherein

(i) at least one instruction of said second instruction set specifies a multi-step operation that requires a plurality of operations that may be specified by instructions of said first instruction set in order to be performed by said processor core; and

(ii) said instruction translator is operable to generate a sequence of translator output signals to control said processor core to perform said multi-step operation.

23. Apparatus as claimed in claim 22 and any one of claims 2 to 12.

24. A method of processing data using a processor core operable to execute operations as specified by instructions of a first instruction set, said processor core having an instruction pipeline into which instructions to be executed are fetched from a memory and along which instructions progress, said method comprising the steps of:

(i) fetching instructions into said instruction pipeline; and

(ii) translating fetched instructions of a second instruction set into translator output signals corresponding to instructions of said first instruction set using an instruction translator within said instruction pipeline; wherein

(iii) said instructions of said second instruction set are variable length instructions; said instruction translator is within said instruction pipeline and translates instructions of said second instruction set that have been fetched into a fetch stage of said instruction pipeline from said memory; and

(iv) said fetch stage of said instruction pipeline includes an instruction buffer holding at least a current instruction word and a next instruction word fetched from said memory such that if a variable length instruction of said second instruction set starts within said current instruction word and extends into said next instruction word, then said next instruction word is available within said pipeline for translation by said instruction translator without requiring a further fetch operation.

25. A computer program product holding a computer program for controlling a computer to perform the method of claim 24.

26. Apparatus for data processing substantially as hereinbefore described with reference to the accompanying drawings.

27. A method of data processing substantially as hereinbefore described with reference to the accompanying drawings.

28. A computer program product holding a computer program for controlling a computer to perform a method substantially as hereinbefore described with reference to the accompanying drawings.